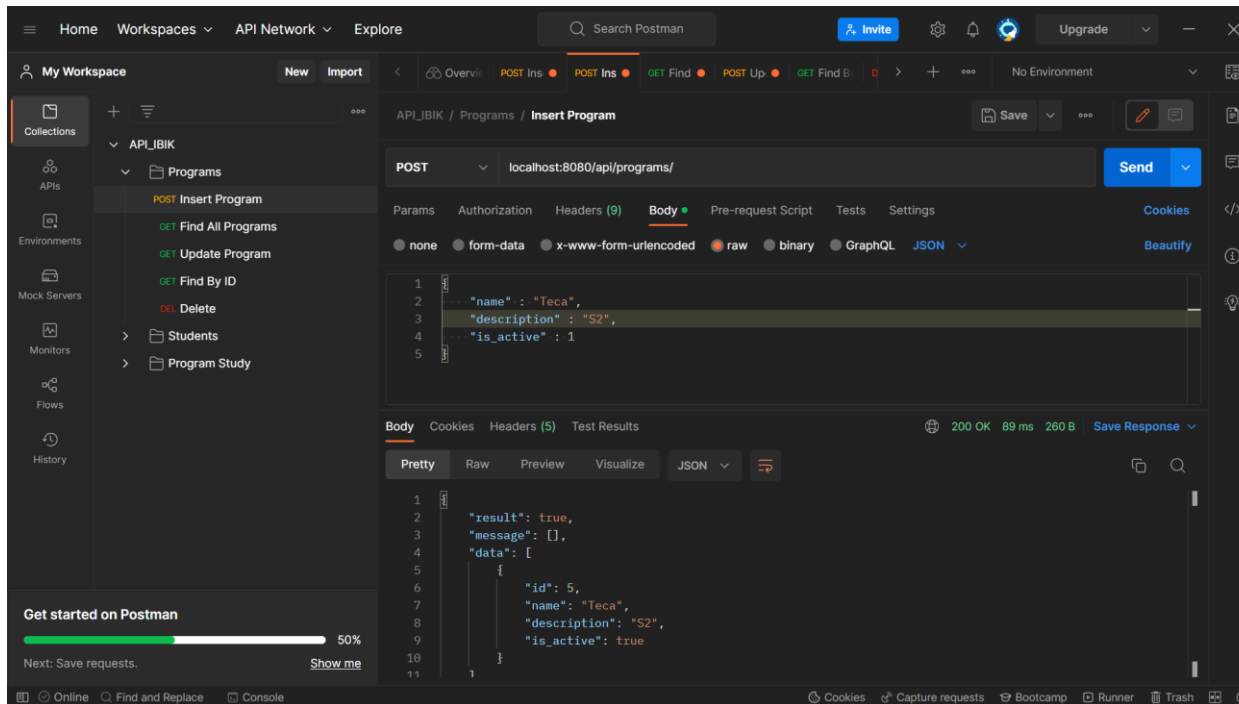
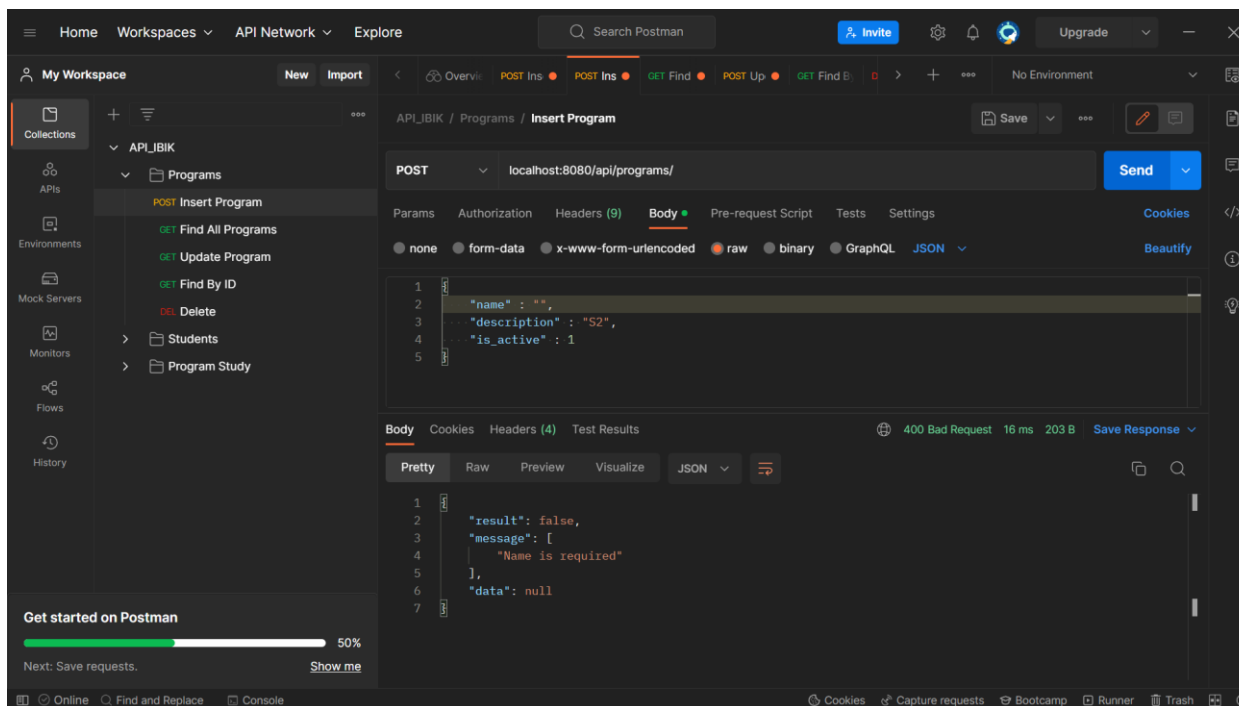


## PROGRAMS



PROGRAMS: Create (Insert Program) Success



PROGRAMS: Create (Insert Program) Error

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection 'API\_IBIK' with a folder 'Programs' containing several endpoints. The selected endpoint is 'GET Find All Programs'. The main panel shows the request details for 'GET localhost:8080/api/programs/'. The response is displayed in the 'Body' tab, showing a JSON object with 'result': true, 'message': [], and 'data': an array of program details. The status is '200 OK' with a response time of '35 ms' and a size of '469 B'.

API\_IBIK / Programs / Find All Programs

GET localhost:8080/api/programs/

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results 200 OK 35 ms 469 B Save Response

Pretty Raw Preview Visualize JSON

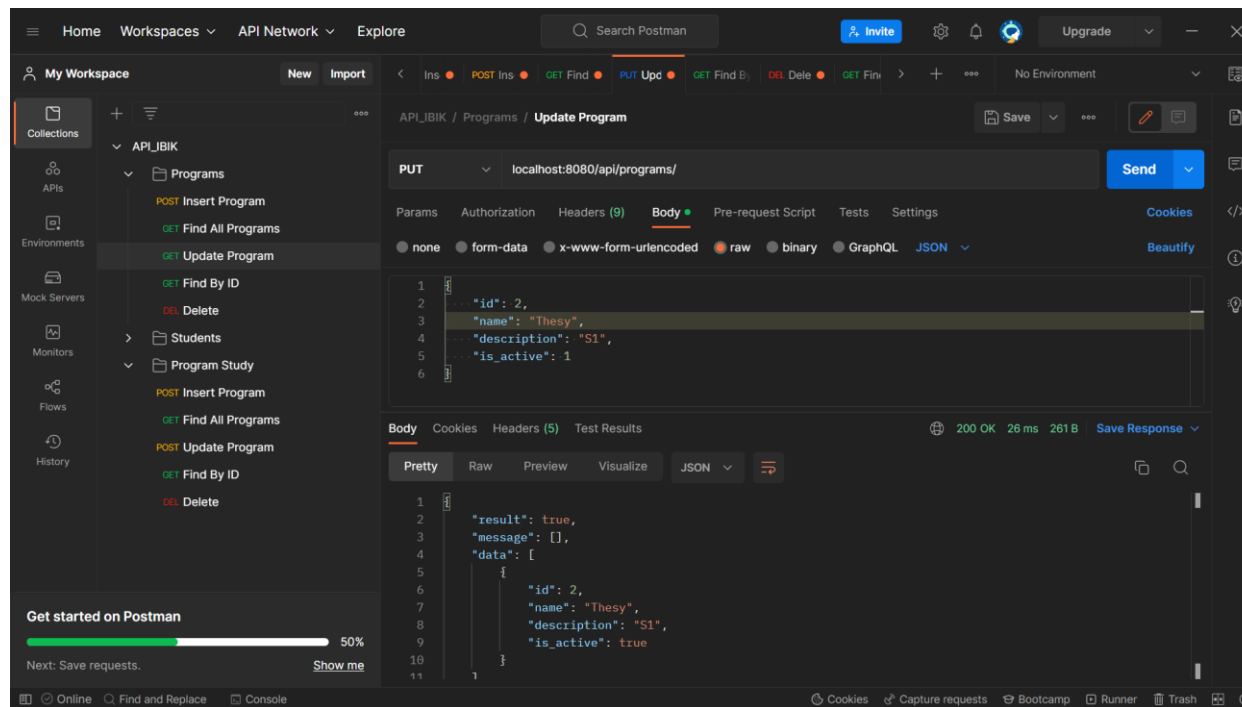
```
1 {
2   "result": true,
3   "message": [],
4   "data": [
5     {
6       "id": 2,
7       "name": "Thesya",
8       "description": "S1",
9       "is_active": true
10    },
11  ]
}
```

Get started on Postman

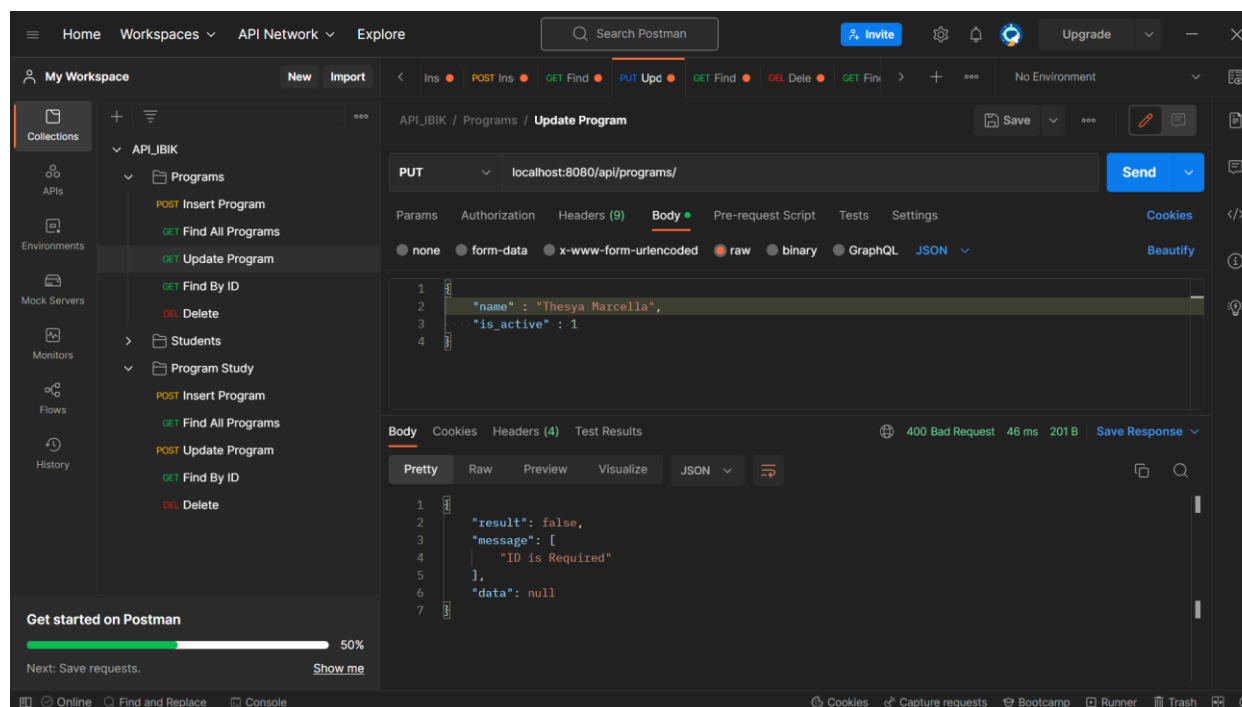
Next: Save requests. Show me

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

PROGRAMS: Retrieve (Find All Program) Success



PROGRAMS: Update Success



PROGRAMS: Update Error

The screenshot shows the Postman interface with a workspace named 'API\_IBIK'. The 'Programs' collection is expanded, and the 'Find By ID' endpoint is selected. The request is a GET method to 'localhost:8080/api/programs/2'. The response is a 200 OK status with a response time of 70 ms and a body of 262 B. The response body is displayed in JSON format:

```
1 {
2   "result": true,
3   "message": [],
4   "data": [
5     {
6       "id": 2,
7       "name": "Thesya",
8       "description": "S1",
9       "is_active": true
10    }
11  ]
12 }
```

PROGRAMS: Retrieve (Find by Id) Success

The screenshot shows the Postman interface with the same workspace and collection. The 'Find By ID' endpoint is selected, but the request is now a GET method to 'localhost:8080/api/programs/20'. The response is a 400 Bad Request status with a response time of 32 ms and a body of 203 B. The response body is displayed in JSON format:

```
1 {
2   "result": false,
3   "message": [
4     "No value present"
5   ],
6   "data": null
7 }
```

PROGRAMS: Retrieve (Find by Id) Error

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection 'API\_IBIK' with a folder 'Programs' containing several endpoints. The 'Delete' endpoint is selected. The main panel shows the request details for 'DELETE localhost:8080/api/programs/3'. The response is a 200 OK status with a response time of 86 ms and a body of JSON data: 

```
{  "result": true,  "message": [    "Successfully Remove"  ],  "data": null}
```

Get started on Postman  
Next: Save requests. [Show me](#)

PROGRAMS: Delete Success

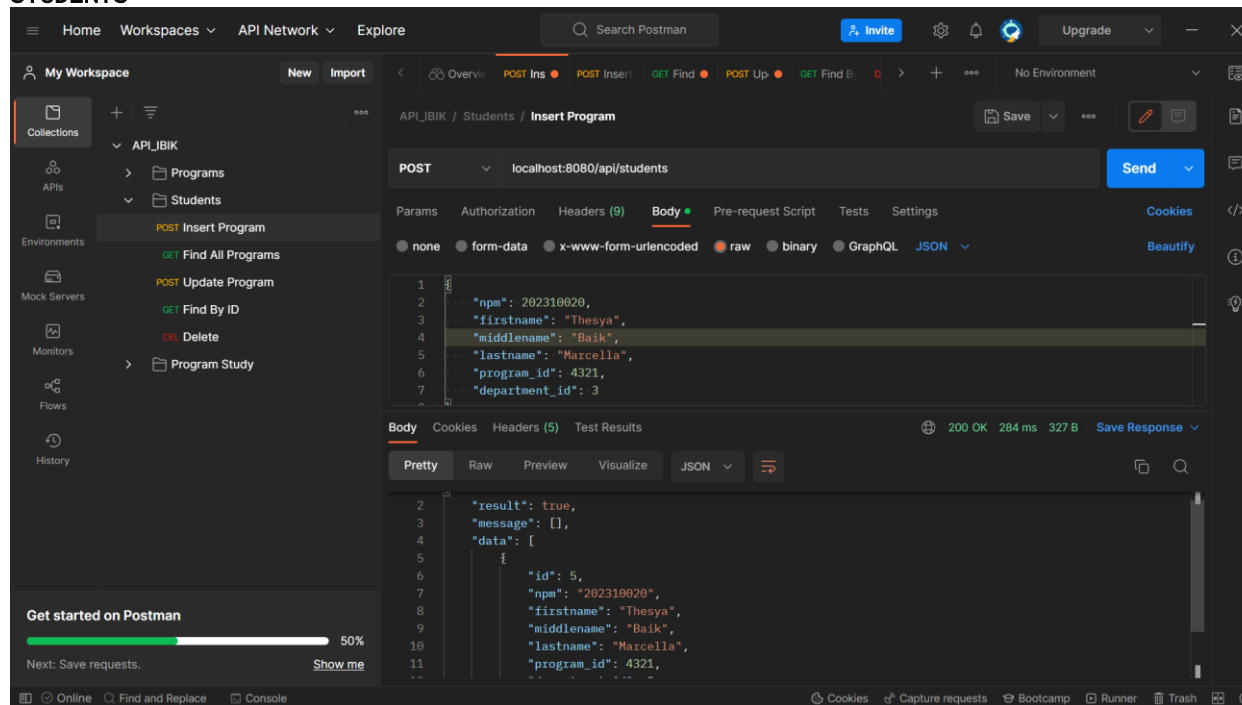
The screenshot shows the Postman interface with the same workspace. The 'Delete' endpoint is selected, but the URL is 'localhost:8080/api/programs/13'. The response is a 400 Bad Request status with a response time of 39 ms and a body of JSON data: 

```
{  "result": false,  "message": [    "No class com.ibik.academicroservices.academicroservices.programs.Programs entity with id 13 exists!"  ],  "data": null}
```

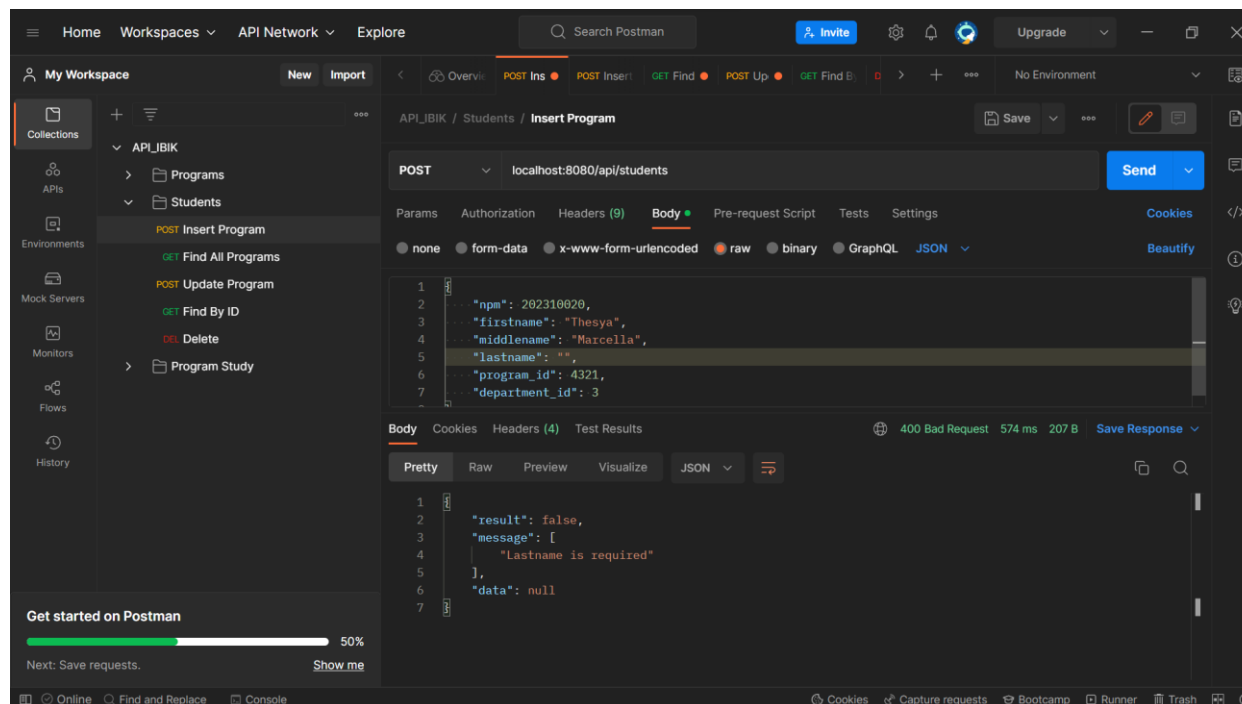
Get started on Postman  
Next: Save requests. [Show me](#)

PROGRAMS: Delete Error

## STUDENTS



STUDENTS: Create (Insert Program) Success



STUDENTS: Create (Insert Program) Error

The screenshot displays the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection 'API\_IBIK' containing a folder 'Students' with several endpoints. The main panel shows a GET request to 'localhost:8080/api/students' which has been successfully executed. The response is a JSON object indicating success and containing a list of student data.

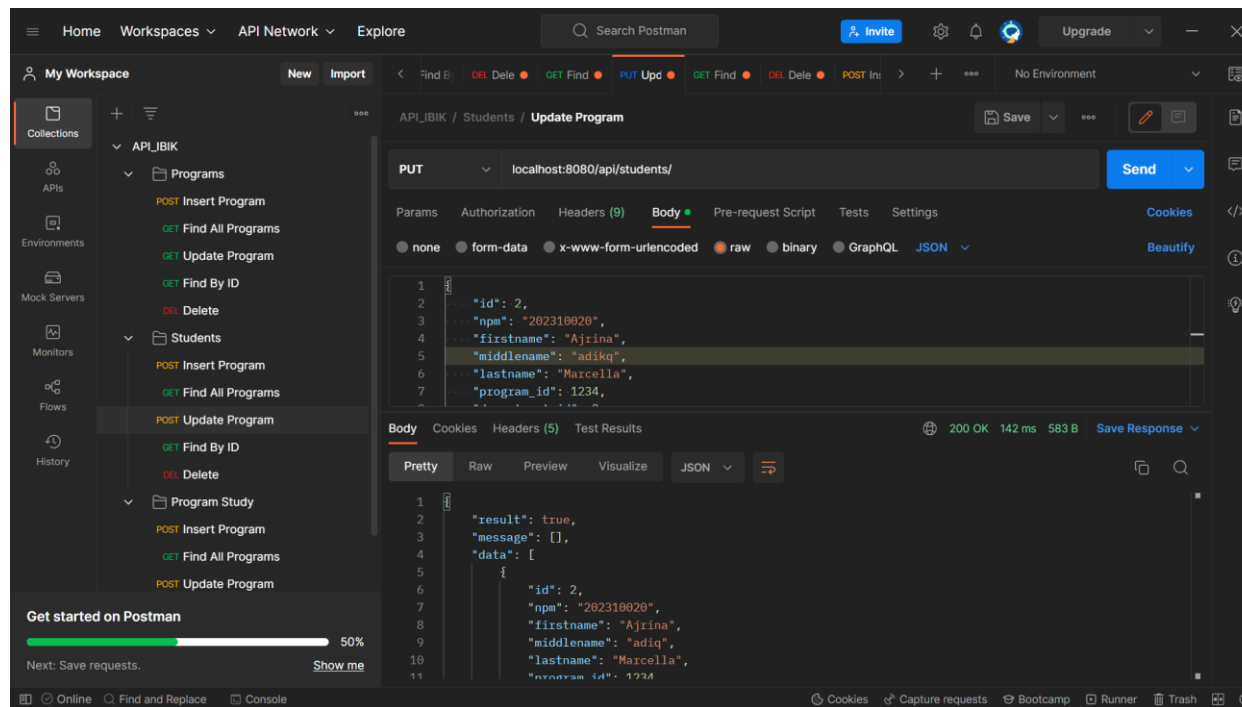
**Request Details:**

- Method: GET
- URL: localhost:8080/api/students
- Status: 200 OK
- Time: 308 ms
- Size: 687 B

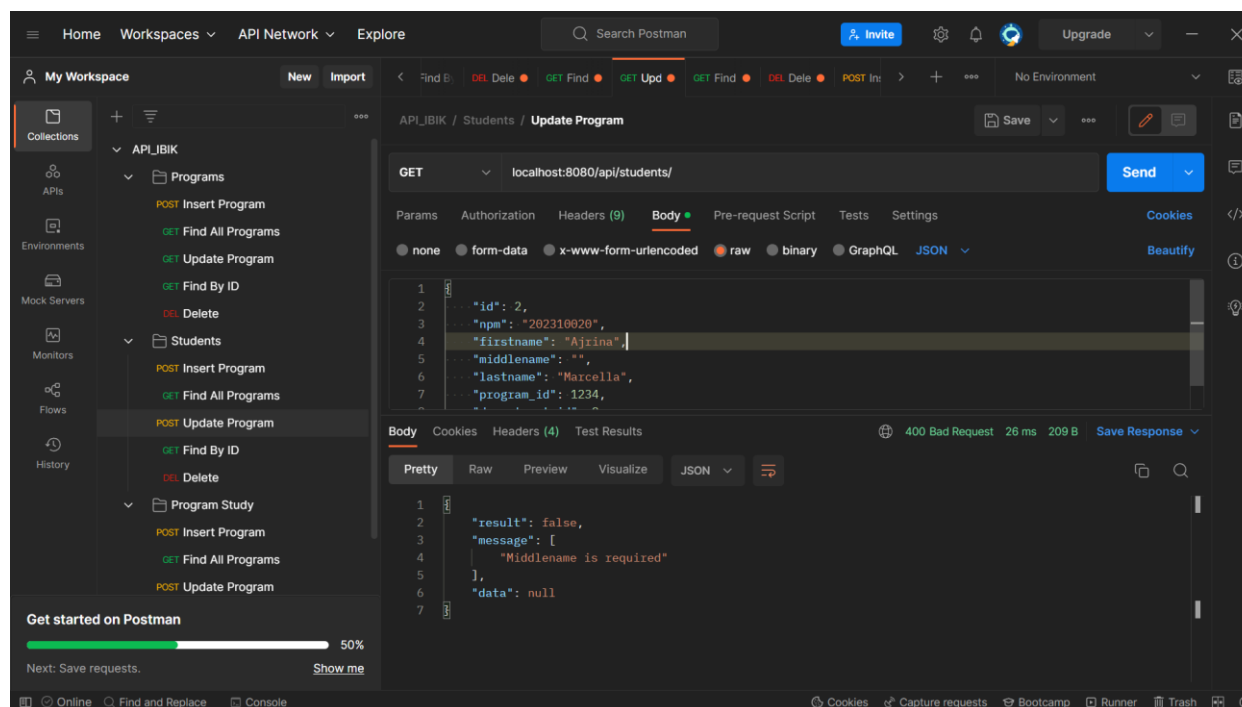
**Response Body (JSON):**

```
1 {
2   "result": true,
3   "message": [],
4   "data": [
5     {
6       "id": 1,
7       "npm": null,
8       "firstname": null,
9       "middlename": null,
10      "lastname": null,
11      "program_id": 0,
12      "department_id": 0
13    },
14    {
15      "id": 2,
16      "npm": "202310020",
17      "firstname": "Ajrina",
18      "middlename": null,
19      "lastname": null,
20      "program_id": 0,
21      "department_id": 0
22    }
23  ]
24 }
```

STUDENTS: Retrieve (Find All Program) Success



STUDENTS: Update Success



STUDENTS: Update Error



The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection 'API\_IBIK' with a folder 'Students' containing a 'GET Find By ID' request. The main panel shows the details of this request, which is a GET request to 'localhost:8080/api/students/2'. The response is a 200 OK status with a 40 ms response time and 327 B of data. The response body is in JSON format, showing a successful result with a message and data.

```
1 {
2   "result": true,
3   "message": [],
4   "data": [
5     {
6       "id": 2,
7       "npm": "202310020",
8       "firstname": "Ajrina",
9       "middlename": "adiq",
10      "lastname": "Marcella",
11      "program id": 1234
12    }
13  ]
14 }
```

STUDENTS: Retrieve (Find by Id) Success

The screenshot shows the Postman interface with the same workspace. The 'GET Find By ID' request is now set to 'localhost:8080/api/students/9'. The response is a 400 Bad Request status with a 27 ms response time and 203 B of data. The response body is in JSON format, showing a failed result with a message indicating 'No value present'.

```
1 {
2   "result": false,
3   "message": [
4     "No value present"
5   ],
6   "data": null
7 }
```

STUDENTS: Retrieve (Find by Id) Error

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar shows a collection 'API\_IBIK' with a sub-collection 'Students'. The 'Delete' endpoint is selected. The request is a DELETE method to 'localhost:8080/api/students/1'. The response is a 200 OK status with a 57 ms response time and 225 B of data. The response body is in JSON format, showing a successful deletion.

**DELETE** `localhost:8080/api/students/1` **Send**

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results **200 OK** 57 ms 225 B **Save Response**

Pretty Raw Preview Visualize JSON **JSON**

```
1 {
2   "result": true,
3   "message": [
4     "Successfully Remove"
5   ],
6   "data": null
7 }
```

Get started on Postman

Next: Save requests. [Show me](#)

### STUDENTS: Delete Success

The screenshot shows the Postman interface with the same workspace. The 'Delete' endpoint is selected, but the URL is 'localhost:8080/api/students/3'. The response is a 400 Bad Request status with a 44 ms response time and 281 B of data. The response body is in JSON format, indicating that no entity with id 3 exists.

**DELETE** `localhost:8080/api/students/3` **Send**

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (4) Test Results **400 Bad Request** 44 ms 281 B **Save Response**

Pretty Raw Preview Visualize JSON **JSON**

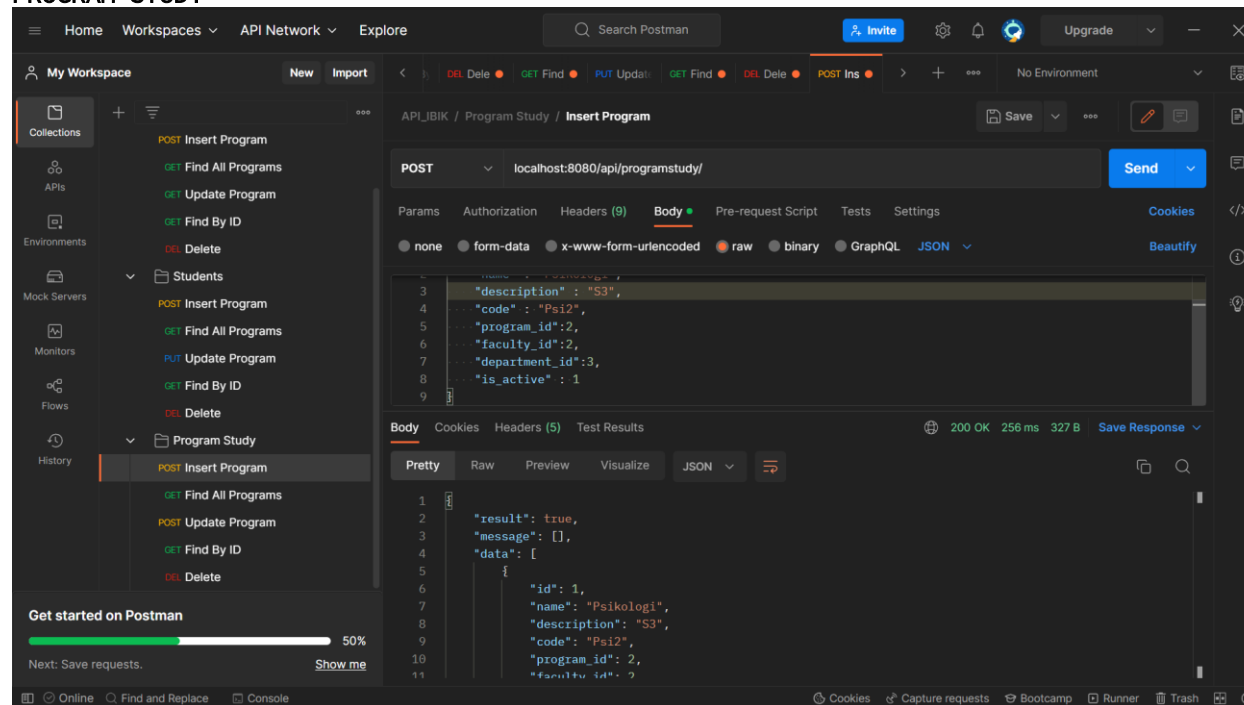
```
1 {
2   "result": false,
3   "message": [
4     "No class com.ibik.academicsservices.academicsservices.students.Students entity with id 3 exists!"
5   ],
6   "data": null
7 }
```

Get started on Postman

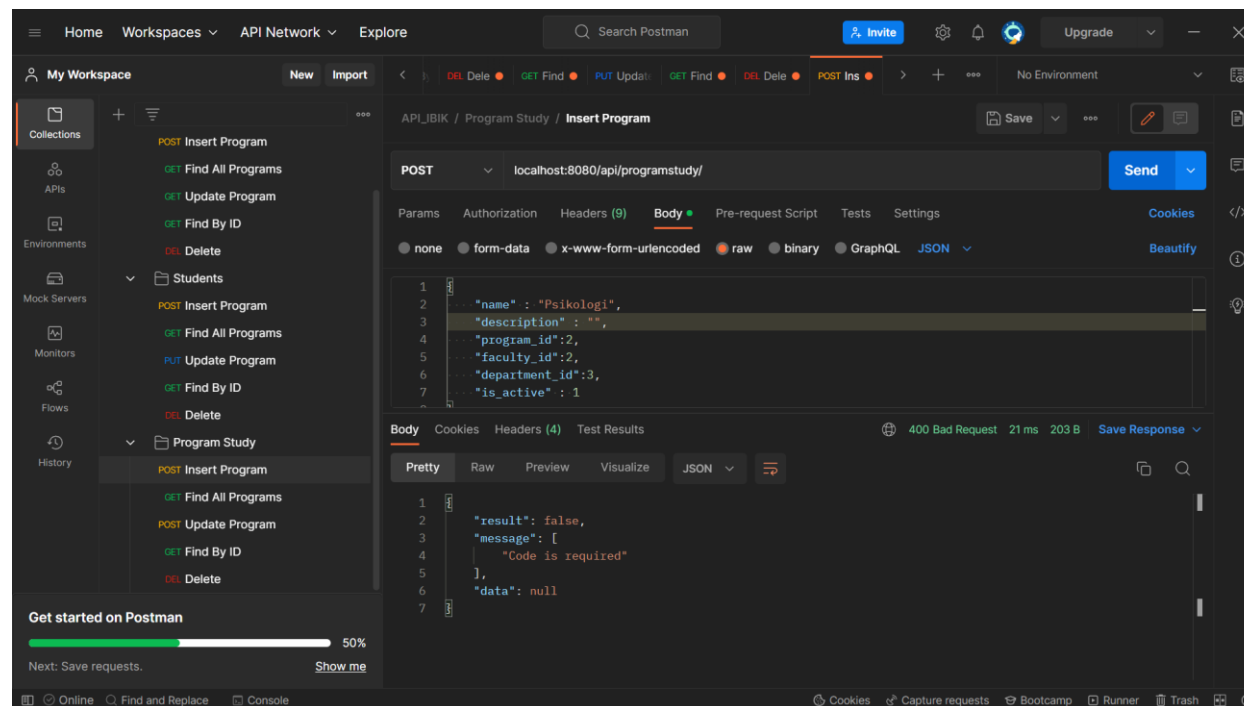
Next: Save requests. [Show me](#)

### STUDENTS: Delete Error

## PROGRAM STUDY



PROGRAMSTUDY: Create (Insert Program) Success



PROGRAMSTUDY: Create (Insert Program) Error

The screenshot shows the Postman interface with a workspace named 'My Workspace'. The left sidebar lists various API endpoints under 'Program Study', including 'POST Insert Program', 'GET Find All Programs', 'GET Update Program', 'GET Find By ID', and 'DELETE Delete'. The main panel displays a GET request to 'localhost:8080/api/programstudy/'. The response is a 200 OK status with a response time of 27 ms and a body size of 451 B. The response body is shown in JSON format, indicating a successful retrieval of program data.

API\_IBIK / Program Study / Find All Programs

GET localhost:8080/api/programstudy/

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results 200 OK 27 ms 451 B Save Response

Pretty Raw Preview Visualize JSON

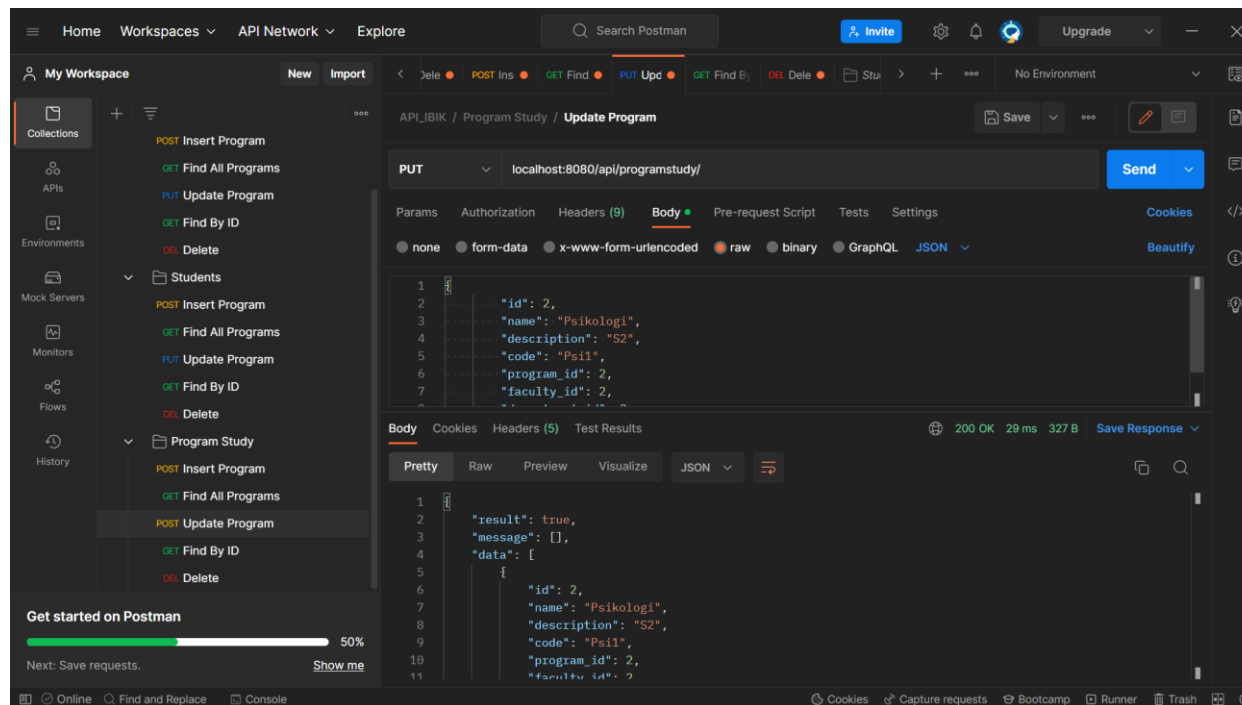
```
1 {
2   "result": true,
3   "message": [],
4   "data": [
5     {
6       "id": 1,
7       "name": "Psikologi",
8       "description": "S3",
9       "code": "Psi2",
10      "program_id": 2,
11      "faculty_id": 2
12    }
13  ]
14 }
```

Get started on Postman

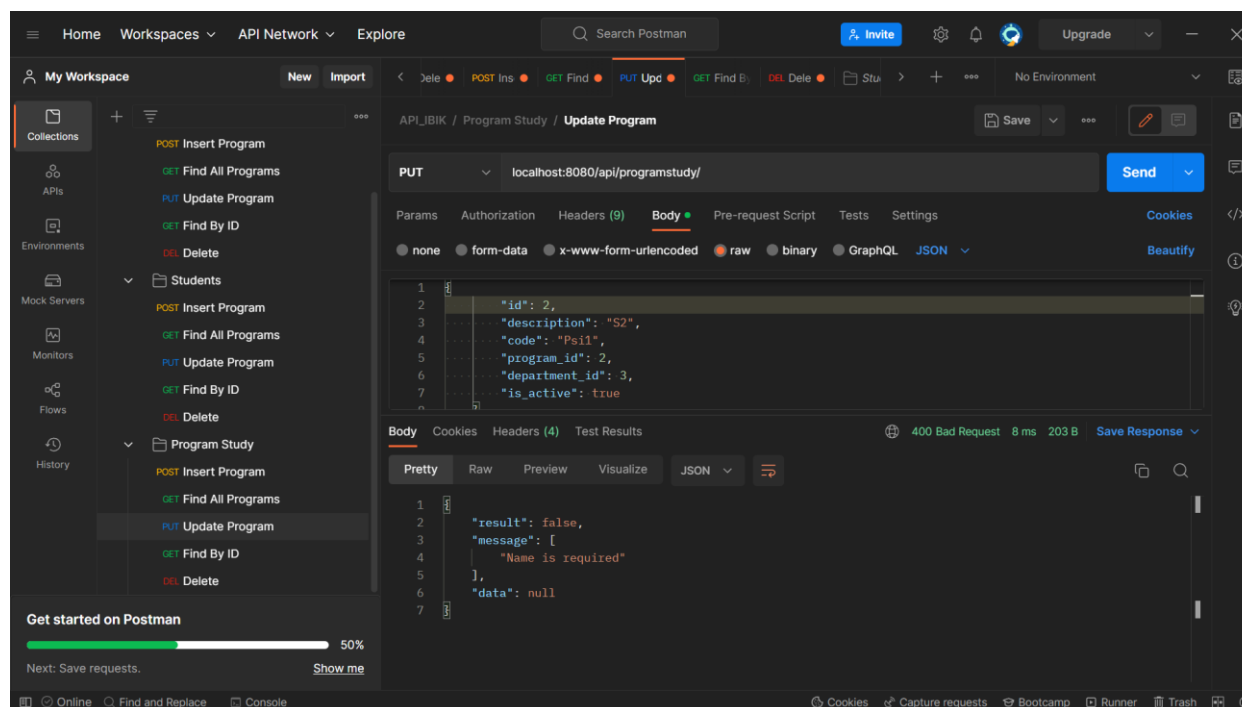
Next: Save requests. Show me

Online Find and Replace Console Cookies Capture requests Bootcamp Runner Trash

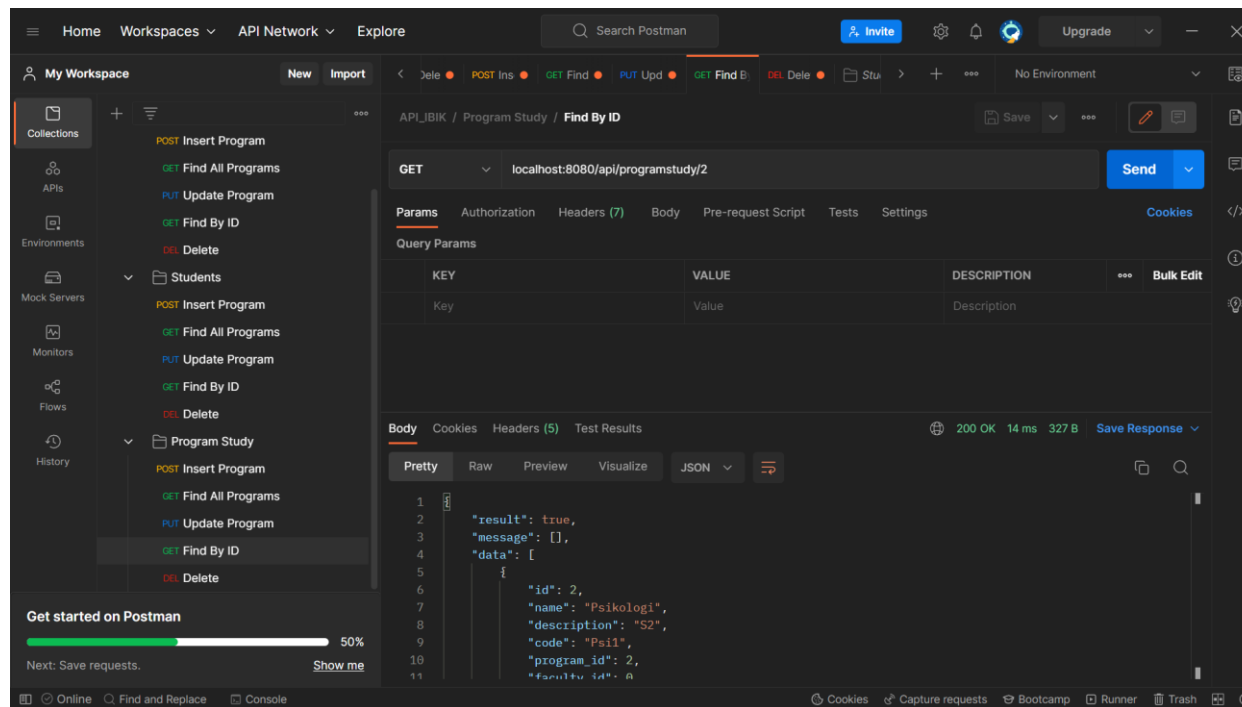
PROGRAMSTUDY: Retrieve (Find All Program) Success



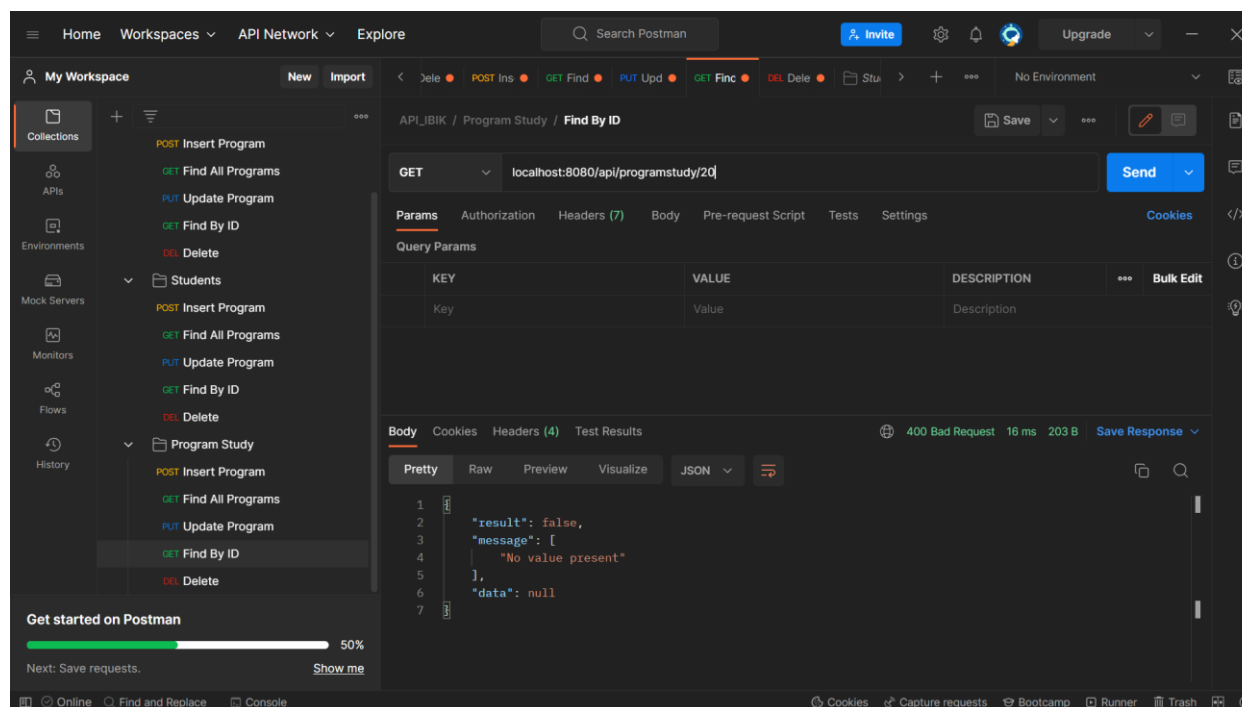
PROGRAMSTUDY: Update Success



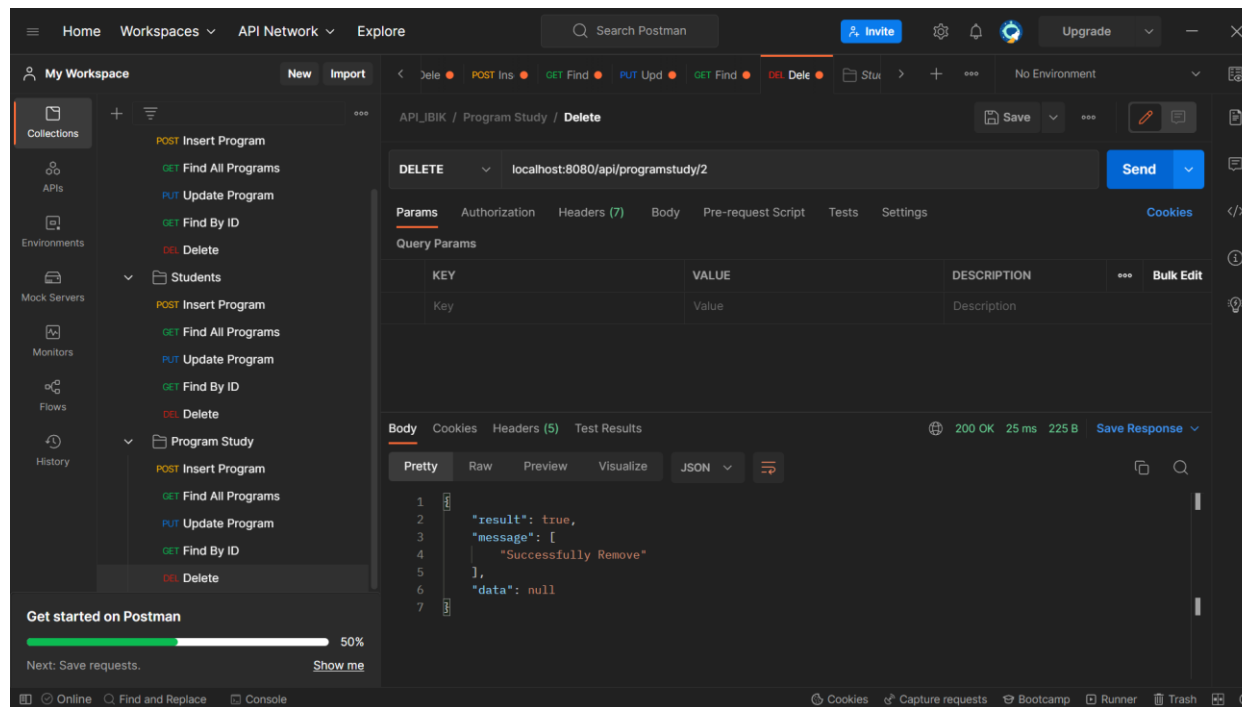
PROGRAMSTUDY: Update Error



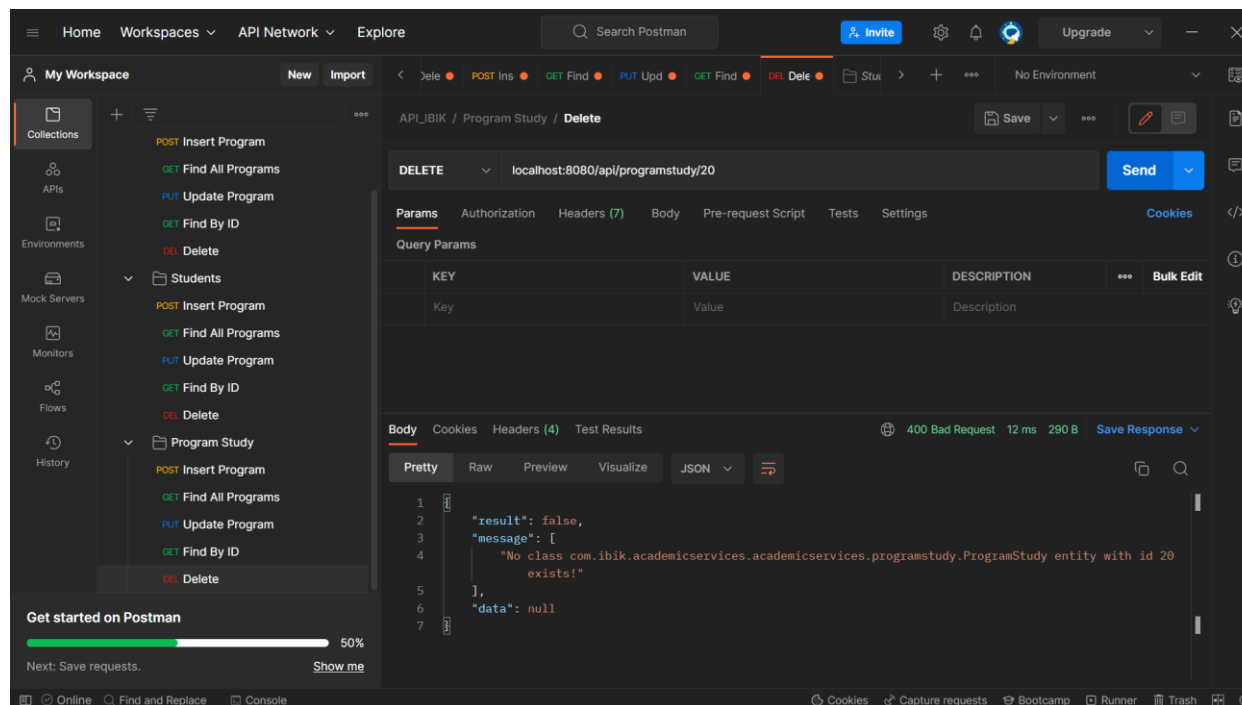
PROGRAMSTUDY: Retrieve (Find by Id) Success



PROGRAMSTUDY: Retrieve (Find by Id) Error



PROGRAMSTUDY: Delete Success



PROGRAMSTUDY: Delete Error